NAG Toolbox

nag_mv_gaussian_mixture (g03ga)

1 Purpose

nag_mv_gaussian_mixture (g03ga) performs a mixture of Normals (Gaussians) for a given (co)variance structure.

2 Syntax

```
[prob, niter, w, g, s, f, loglik, ifail] = nag_mv_gaussian_mixture(x, isx, nvar, ng, sopt, sds, tol, 'n', n, 'm', m, 'prob', prob, 'niter', niter, 'riter', riter)
[prob, niter, w, g, s, f, loglik, ifail] = g03ga(x, isx, nvar, ng, sopt, sds, tol, 'n', n, 'm', m, 'prob', prob, 'niter', niter, 'riter', riter)
```

3 Description

A Normal (Gaussian) mixture model is a weighted sum of k group Normal densities given by,

$$p(x \mid w, \mu, \Sigma) = \sum_{j=1}^{k} w_j g\left(x \mid \mu_j, \Sigma_j\right), \quad x \in \mathbb{R}^p$$

where:

x is a p-dimensional object of interest;

w_j is the mixture weight for the jth group and $\sum_{j=1}^{k} w_j = 1$;

μ_j is a p-dimensional vector of means for the jth group;

Σ_j is the covariance structure for the jth group;

$g(\cdot)$ is the p-variate Normal density:

$$g\left(x \mid \mu_j, \Sigma_j\right) = \frac{1}{(2\pi)^{p/2}\left|\Sigma_j\right|^{1/2}} \exp\left[-\frac{1}{2}\left(x-\mu_j\right)\Sigma_j^{-1}\left(x-\mu_j\right)^{\mathrm{T}}\right].$$

Optionally, the (co)variance structure may be pooled (common to all groups) or calculated for each group, and may be full or diagonal.

4 References

Hartigan J A (1975) *Clustering Algorithms* Wiley

5 Parameters

5.1 Compulsory Input Parameters

1: **x**$(ldx, \mathbf{m})$ – REAL (KIND=nag_wp) array

ldx, the first dimension of the array, must satisfy the constraint $ldx \geq \mathbf{n}$.

x(i, j) must contain the value of the jth variable for the ith object, for $i = 1, 2, \ldots, n$ and $j = 1, 2, \ldots, \mathbf{m}$.

2: **isx**(**m**) – INTEGER array

If **nvar** = **m** all available variables are included in the model and **isx** is not referenced; otherwise the jth variable will be included in the analysis if **isx**$(j) = 1$ and excluded if **isx**$(j) = 0$, for $j = 1, 2, \ldots,$ **m**.

Constraint: if **nvar** $\neq$ **m**, **isx**$(j) = 1$ for **nvar** values of j and **isx**$(j) = 0$ for the remaining **m** $-$ **nvar** values of j, for $j = 1, 2, \ldots,$ **m**.

3: **nvar** – INTEGER

p, the number of variables included in the calculations.

Constraint: $1 \leq$ **nvar** $\leq$ **m**.

4: **ng** – INTEGER

k, the number of groups in the mixture model.

Constraint: **ng** ≥ 1.

5: **sopt** – INTEGER

Determines the (co)variance structure:

sopt $= 1$
Groupwise covariance matrices.

sopt $= 2$
Pooled covariance matrix.

sopt $= 3$
Groupwise variances.

sopt $= 4$
Pooled variances.

sopt $= 5$
Overall variance.

Constraint: **sopt** $= 1$, 2, 3, 4 or 5.

6: **sds** – INTEGER

The second dimension of the (co)variance structure **s**.

Constraints:

if **sopt** $= 1$ or 2, **sds** must be at least **nvar**;
if **sopt** $= 3$, **sds** must be at least **ng**;
if **sopt** $= 4$ or 5, **sds** must be at least 1.

7: **tol** – REAL (KIND=nag_wp)

Iterations cease the first time an improvement in log-likelihood is less than **tol**. If **tol** ≤ 0 a value of 10^{-3} is used.

5.2 Optional Input Parameters

1: **n** – INTEGER

Default: the first dimension of the arrays **x**, **prob**. (An error is raised if these dimensions are not equal.)

n, the number of objects. There must be more objects than parameters in the model.

Constraints:

if **sopt** $= 1$, **n** $>$ **ng** $\times$ (**nvar** $\times$ **nvar** $+$ **nvar**);
if **sopt** $= 2$, **n** $>$ **nvar** $\times$ (**ng** $+$ **nvar**);
if **sopt** $= 3$, **n** $> 2 \times$ **ng** $\times$ **nvar**;
if **sopt** $= 4$, **n** $>$ **nvar** $\times$ (**ng** $+ 1$);
if **sopt** $= 5$, **n** $>$ **nvar** $\times$ **ng** $+ 1$.

2: **m** – INTEGER

Default: the dimension of the array **isx** and the second dimension of the array **x**. (An error is raised if these dimensions are not equal.)

The total number of variables in array **x**.

Constraint: **m** ≥ 1.

3: **prob**$(lprob, \mathbf{ng})$ – REAL (KIND=nag_wp) array

If $popt \neq 1$, **prob**(i, j) is the probability that the ith object belongs to the jth group. (These probabilities are normalised internally.)

4: **niter** – INTEGER

Default: 15

The maximum number of iterations.

Constraint: **niter** ≥ 1.

5: **riter** – INTEGER

Default: 5

If **riter** > 0, membership probabilities are rounded to 0.0 or 1.0 after the completion of every **riter** iterations.

5.3 Output Parameters

1: **prob**$(lprob, \mathbf{ng})$ – REAL (KIND=nag_wp) array

prob(i, j) is the probability of membership of the ith object to the jth group for the fitted model.

2: **niter** – INTEGER

Default: 15

The number of completed iterations.

3: **w**$(\mathbf{ng})$ – REAL (KIND=nag_wp) array

w_j, the mixing probability for the jth group.

4: **g**$(\mathbf{nvar}, \mathbf{ng})$ – REAL (KIND=nag_wp) array

g(i, j) gives the estimated mean of the ith variable in the jth group.

5: **s**$(lds, \mathbf{sds}, :)$ – REAL (KIND=nag_wp) array

The last dimension of the array **s** will be **ng** if **sopt** $= 1$ and 1 otherwise

If **sopt** $= 1$, **s**(i, j, k) gives the (i, j)th element of the kth group.

If **sopt** $= 2$, **s**$(i, j, 1)$ gives the (i, j)th element of the pooled covariance.

If **sopt** $= 3$, **s**$(j, k, 1)$ gives the jth variance in the kth group.

If **sopt** $= 4$, **s**$(j, 1, 1)$ gives the jth pooled variance.

If **sopt** $= 5$, $\mathbf{s}(1,1,1)$ gives the overall variance.

6: **f**(**n**, **ng**) – REAL (KIND=nag_wp) array

$\mathbf{f}(i,j)$ gives the p-variate Normal (Gaussian) density of the ith object in the jth group.

7: **loglik** – REAL (KIND=nag_wp)

The log-likelihood for the fitted mixture model.

8: **ifail** – INTEGER

ifail $= 0$ unless the function detects an error (see Section 5).

6 Error Indicators and Warnings

Errors or warnings detected by the function:

ifail $= 1$

Constraint: **n** $> p$, the number of parameters, i.e., too few objects have been supplied for the model.

ifail $= 2$

Constraint: **m** ≥ 1.

ifail $= 4$

Constraint: $ldx \geq$ **n**.

ifail $= 5$

Constraint: $1 \leq$ **nvar** $\leq$ **m**.

ifail $= 6$

On entry, **nvar** $\neq$ **m** and **isx** is invalid.

ifail $= 7$

Constraint: **ng** ≥ 1.

ifail $= 8$

On entry, $popt \neq 1$ or 2.

ifail $= 9$

On entry, row $\langle value \rangle$ of supplied **prob** does not sum to 1.

ifail $= 10$

Constraint: $lprob \geq$ **n**.

ifail $= 11$

Constraint: **niter** ≥ 1.

ifail $= 16$

On entry, **sopt** < 1 or **sopt** > 5.

ifail $= 18$

On entry, $lds = \langle value \rangle$ was invalid.

ifail $= 19$

On entry, **sds** $= \langle value \rangle$ was invalid.

ifail $= 44$

A covariance matrix is not positive definite, try a different initial allocation.

ifail $= 45$

An iteration cannot continue due to an empty group, try a different initial allocation.

ifail $= -99$

An unexpected error has been triggered by this routine. Please contact NAG.

ifail $= -399$

Your licence key may have expired or may not have been installed correctly.

ifail $= -999$

Dynamic memory allocation failed.

7 Accuracy

Not applicable.

8 Further Comments

None.

9 Example

This example fits a Gaussian mixture model with pooled covariance structure to New Haven schools test data, see Table 5.1 (p. 118) in Hartigan (1975).

9.1 Program Text

```
function g03ga_example

fprintf('g03ga example results\n\n');

x = [2.7, 3.2, 4.5, 4.8;
     3.9, 3.8, 5.9, 6.2;
     4.8, 4.1, 6.8, 5.5;
     3.1, 3.5, 4.3, 4.6;
     3.4, 3.7, 5.1, 5.6;
     3.1, 3.4, 4.1, 4.7;
     4.6, 4.4, 6.6, 6.1;
     3.1, 3.3, 4.0, 4.9;
     3.8, 3.7, 4.7, 4.9;
     5.2, 4.9, 8.2, 6.9;
     3.9, 3.8, 5.2, 5.4;
     4.1, 4.0, 5.6, 5.6;
     5.7, 5.1, 7.0, 6.3;
     3.0, 3.2, 4.5, 5.0;
     2.9, 3.3, 4.5, 5.1;
     3.4, 3.3, 4.4, 5.0;
     4.0, 4.2, 5.2, 5.4;
     3.0, 3.0, 4.6, 5.0;
     4.0, 4.1, 5.9, 5.8;
     3.0, 3.2, 4.4, 5.1;
     3.6, 3.6, 5.3, 5.4;
     3.1, 3.2, 4.6, 5.0;
     3.2, 3.3, 5.4, 5.3;
```

```
     3.0, 3.4, 4.2, 4.7;
     3.8, 4.0, 6.9, 6.7];

[m,n] = size(x);

ng    = nag_int(2);
prob = zeros(m,ng);
prob(1:12,1) = 1;
prob(13:m,2) = 1;
isx = zeros(n, 1, nag_int_name);

nvar  = nag_int(n);
sopt  = nag_int(2);
sds   = nvar;
tol   = 0;
[prob, niter, w, g, s, f, loglik, ifail] = ...
  g03ga( ...
  x, isx, nvar, ng, sopt, sds, tol, 'prob', prob);

mtitle = 'Mixing proportions';
matrix = 'General';
diag   = ' ';
[ifail] = x04ca( ...
                matrix, diag, w', mtitle);

fprintf('\n');
mtitle = 'Group means';
[ifail] = x04ca( ...
                matrix, diag, g, mtitle);

fprintf('\n');
mtitle = 'Pooled Variance-covariance matrix';
[ifail] = x04ca( ...
                matrix, diag, s, mtitle);

fprintf('\n');
mtitle = 'Densities';
[ifail] = x04ca( ...
                matrix, diag, f, mtitle);

fprintf('\n');
mtitle = 'Membership probabilities';
[ifail] = x04ca( ...
                matrix, diag, prob, mtitle);
fprintf('\nNumber of iterations = %5d\n', niter);
fprintf(  'Log-likelihood       = %10.4f\n:', loglik);
```

9.2 Program Results

```
     g03ga example results

 Mixing proportions
         1       2
 1   0.4798  0.5202

 Group means
             1          2
 1      4.0041     3.3350
 2      3.9949     3.4434
 3      5.5894     4.9870
 4      5.4432     5.3602

 Pooled Variance-covariance matrix
             1          2          3          4
 1      0.4539     0.2891     0.6075     0.3413
 2      0.2891     0.2048     0.4101     0.2490
 3      0.6075     0.4101     1.0648     0.6011
 4      0.3413     0.2490     0.6011     0.3759

 Densities
```

```
                1           2
  1    2.5836E-01   1.1853E-02
  2    3.7065E-07   1.1241E-01
  3    5.3069E-03   1.8080E-06
  4    4.2461E-01   2.8584E-05
  5    5.0387E-02   1.1544E+00
  6    1.1260E+00   7.2224E-02
  7    2.0911E+00   2.1224E-02
  8    5.7856E-03   1.3227E+00
  9    1.1609E+00   2.9411E-02
 10    8.9826E-02   2.4260E-05
 11    3.0170E-01   1.0106E+00
 12    1.2930E+00   3.5422E-01
 13    2.8644E-02   6.7851E-07
 14    2.0759E-02   3.1690E+00
 15    7.6461E-02   1.5231E+00
 16    3.0279E-04   8.4017E-01
 17    5.6101E-01   4.6699E-05
 18    2.6573E-05   6.4442E-01
 19    2.1250E+00   5.1006E-02
 20    8.6822E-04   2.7626E+00
 21    1.9223E-01   2.3971E+00
 22    1.2469E-02   2.8179E+00
 23    1.8389E-02   5.3572E-01
 24    1.2409E+00   9.6489E-03
 25    2.1037E-05   4.8674E-02

 Membership probabilities
                1           2
  1    9.5018E-01   4.9823E-02
  2    3.3259E-06   1.0000E+00
  3    9.9961E-01   3.8664E-04
  4    9.9992E-01   7.9913E-05
  5    3.8999E-02   9.6100E-01
  6    9.3270E-01   6.7295E-02
  7    9.8881E-01   1.1190E-02
  8    4.1252E-03   9.9587E-01
  9    9.7252E-01   2.7479E-02
 10    9.9969E-01   3.0805E-04
 11    2.1722E-01   7.8278E-01
 12    7.6938E-01   2.3062E-01
 13    9.9997E-01   2.6937E-05
 14    6.1133E-03   9.9389E-01
 15    4.4189E-02   9.5581E-01
 16    3.5006E-04   9.9965E-01
 17    9.9990E-01   9.7029E-05
 18    4.0270E-05   9.9996E-01
 19    9.7380E-01   2.6202E-02
 20    3.0204E-04   9.9970E-01
 21    6.9471E-02   9.3053E-01
 22    4.1603E-03   9.9584E-01
 23    3.0839E-02   9.6916E-01
 24    9.9116E-01   8.8421E-03
 25    4.1534E-04   9.9958E-01

Number of iterations =    14
Log-likelihood       =   -29.6831
:
```